SMS SPAM DETECTION

A Project Report

Submitted for Minor Project (6CS191) of 6th Semester for partial fulfillment of the requirements for the award of the degree of

Bachelors of Technology
in
Computer science and Engineering

Submitted by

Raju Kumar	**1706015**
Suraj Kumar	**1706004**
Devendra Singh	**1706022**
Kishan Kumar	**1706013**

Under the Supervision of

Prof. Kakali Chatterjee
Asst. Prof. CSE Department
NIT Patna

Department of Computer Science & Engineering

National Institute of Technology Patna
Patna-800005

Jan-June, 2020

राष्ट्रीय प्रौद्योगिकी संस्थान पटना
NATIONAL INSTITUTE OF TECHNOLOGY PATNA

CERTIFICATE

This is to certify that **Raju Kumar Roll No. 1706015, Suraj Kumar Roll No. 1706004, Devendra Singh Roll No. 1706022, Kishan Kumar Roll No. 1706013** has carried out the Minor project (6CS191) entitled as **" Sms Spam Detection "** during his 6th semester under the supervision of **Prof. Kakali Chatterjee,** Asst. Prof., CSE Department in partial fulfillment of the requirements for the award of Bachelor of Technology degree in the department of Computer Science & Engineering, National Institute of Technology Patna.

……………………………….
Prof. Kakali Chatterjee
Assistant Professor
CSE Department
NIT Patna

……………………………….
Dr. J. P. Singh
Head of Department
CSE Department
NIT Patna

राष्ट्रीय प्रौद्योगिकी संस्थान पटना
NATIONAL INSTITUTE OF TECHNOLOGY PATNA

DECLARATION

We students of 6th semester hereby declare that this project entitled “ **Sms Spam Detection** ” has been carried out by us in the Department of Computer Science and Engineering of National Institute of Technology Patna under the guidance of **Prof. Abhay Kumar**, Department of Computer Science and Engineering, NIT Patna. No part of this project has been submitted for the award of degree or diploma to any other Institute.

Name	**Signature**
Raju Kumar	………………………………
Suraj Kumar	………………………………
Devendra Singh	………………………………
Kishan Kumar	………………………………

Place:	**Date:**
NIT Patna	**………………..**

राष्ट्रीय प्रौद्योगिकी संस्थान पटना
NATIONAL INSTITUTE OF TECHNOLOGY PATNA

ACKNOWLEDGEMENT

We would like to acknowledge and express my deepest gratitude to my mentor **Prof. Kakali Chatterjee**, Assistant Professor, Computer Science & Engineering Department, National Institute of Technology Patna for the valuable guidance, sympathy and co-operation for providing necessary facilities and sources during the entire period of this project.

I wish to convey my sincere gratitude to the Head of Department and all the faculties of Computer Science & Engineering Department who have enlightened me during our studies. The faculties and cooperation received from the technical staff of Department of Computer Science & Engineering is thankfully acknowledged.

1.	Raju Kumar	(Roll No. 1706015)
2.	Suraj Kumar	(Roll No. 1706004)
3.	Devendra Singh	(Roll No. 1706022)
4.	Kishan Kumar	(Roll No. 1706013)

Contents

PageNo.

1. Abstract:

Over recent years, as the popularity of mobile phone devices has increased, Short Message Service (SMS) has grown into a multi-billion dollars industry. At the same time, reduction in the cost of messaging services has resulted in growth in unsolicited commercial advertisements (spams) being sent to mobile phones. SMS spamming is an activity of sending 'unwanted messages' through text messaging or other communication
services; normally using mobile phones. The SMS spam problem can be approached with legal, economic or technical measures.
Nowadays there are many methods for SMS spam detection, ranging from the list-based, statistical algorithm, IP-based and using machine learning. However, an optimum method for SMS spam detection is difficult to find due to issues of SMS length, battery and memory performances.
A database of real SMS Spams from UCI Machine Learning repository is used, and after preprocessing and feature extraction, different machine learning techniques are applied to the database.
Among the wide range of technical measures, Bayesian filters are playing a key role in stopping sms spam. Here, we analyze to what extent Bayesian filtering techniques can be applied to the introduction.

Spams are unwanted messages which can be transmitted over a communication media such as SMS. According to TIME1 and Digital Trends2, 6 billion out of 7 billion people in the world have access to cellphones and it is going to increase to 7.3 billion by 2014. Thus, the number of cellphones will soon outgrow the world population. In 2012, there were more than 6 billion daily Short Message Service (SMS) exchanges over mobile phones just in the US3, and the rate of SMS spams increased by 400 percent. These spam messages not only waste network resources but also increase the cost for mobile phone users and even lead to cyber attacks such as phishing. Therefore, there is a strong need for SMS spam detection.

2. About the Project:

2.1 Introduction:

Objective: To identify text messages/sms as spam or ham(non-spam)

Problem Formulation

In SMS, there are a set of k text messages TM = {tm1 , … , tmk}. Each message is limited to 160 characters consisting of words, number, etc. Messages can be about any topic.

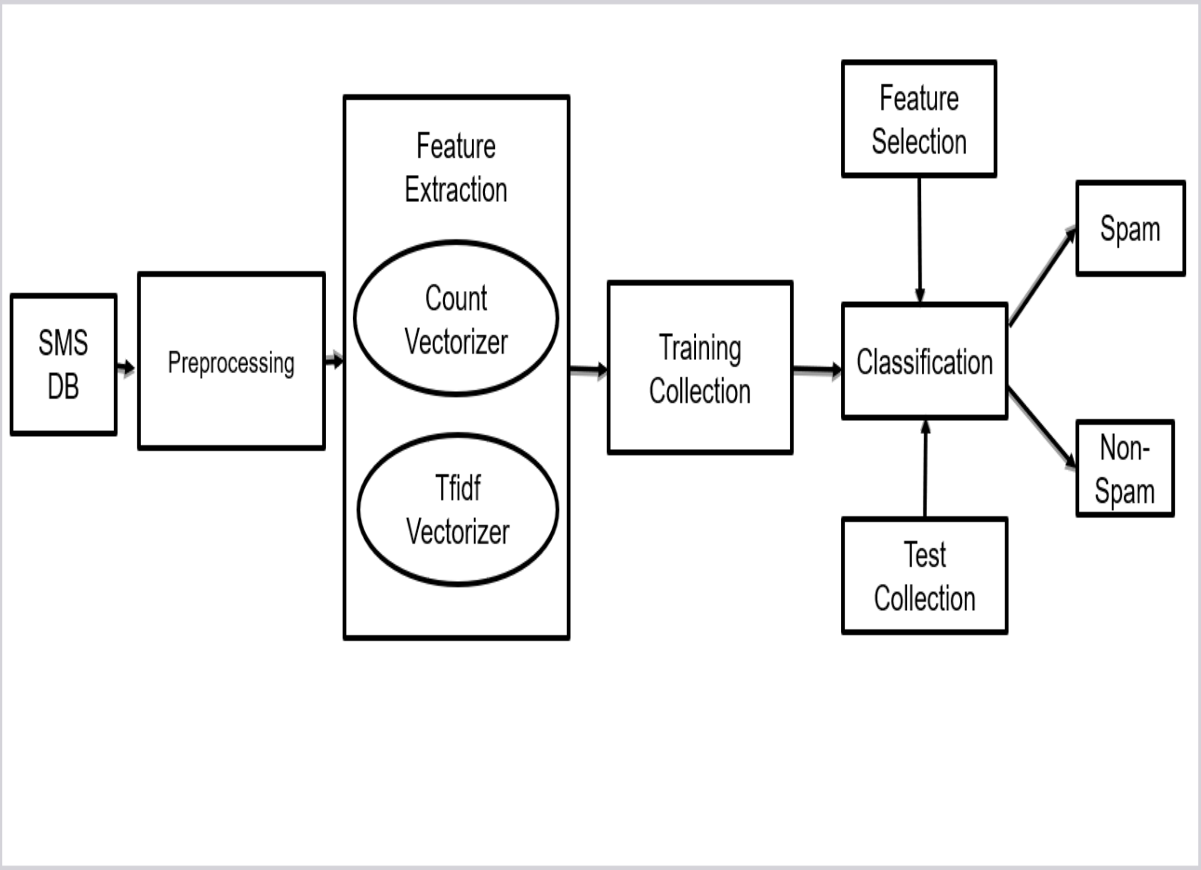
The tasks of SMS spam detection is to predict whether tmi is a spam (A) or non-spam (B) by using a classifier c. The problem is formulated below:

c:tmi → {spam, non − spam}

To support the classification,we need to first extract a set of n features F ={f1 , … ,fn} from TM.

2.2 Framework:

We propose a framework for detecting spam messages in SMS (see Figure). This framework is composed of three major components: preprocessing ,feature extraction and classification. We used stemming and stopwords as our preprocessing .The goal of feature extraction is to transform the input data into a set of features. It is very important in text analysis because it has a direct effect on machine learning techniques to distinguish classes or clusters; moreover, it is hard to find good features in unstructured data. In feature extraction step, we extracted two categories of features which will be introduced in detail in the following sections. In addition, we explored a wide range of classification algorithms from Random Forest to Naive Bayes and different test options to evaluate our proposed framework.



3. Preprocessing:

Data preprocessing is an essential step in building a Machine Learning model and depending on how well the data has been preprocessed; the results are seen.

In NLP, text preprocessing is the first step in the process of building a model.The various text preprocessing steps are:

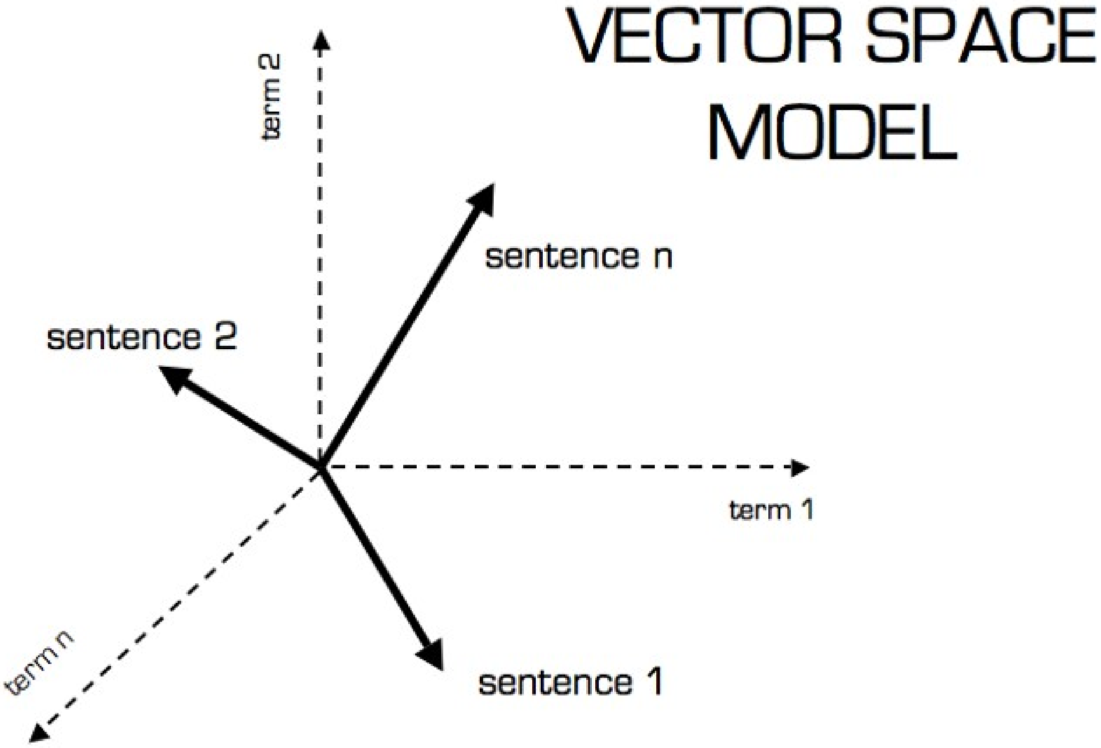
1.Tokenization

2.Lower casing

3.Stop words removal

4.Stemming

5.Lemmatization



In the vector space model, each word/term is an axis/dimension. The text/document is represented as a vector in the multi-dimensional space. The number of unique words means the number of dimensions.

Count Vectorizer :

In order to use textual data for predictive modeling, the text must be parsed to remove certain words – this process is called tokenization. These words need to then be encoded as integers, or floating-point values, for use as inputs in machine learning algorithms. This process is called feature extraction (or vectorization).

Tf-idf vectorizer :

Tf-idf Vectorizer aim to do the same thing, which is to convert a collection of raw documents to a matrix of TF-IDF features. The differences between the two modules can be quite confusing and it’s hard to know when to use which. This article shows you how to correctly use each module, the differences between the two and some guidelines on what to use when.

4. Classification:

Different Algorithms are used for the clssifcation of ham and spam messages.E.x.- Multinomial Naive Bayes Algorithm, Support Vector machine Algorithm, KNN Algorithm,
Muti-Layer Perceptron etc.

The various Algorithms used are as follows :

4.1 Logistic regression:

It is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary).

4.2 Naive Baye's:

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

4.3 K-Nearest Neighbours (KNN) :

k-nearest neighbour can be applied to the classification problems as a simple instance-based learning algorithm. In this method, the label for a test sample is predicted based on the majority vote of its k nearest neighbours. K nearest neighbours is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN has been used in statistical estimation and pattern recognition already in the beginning of 1970's as a non-parametric technique.

4.4 Random Forest

Random forests is an averaging ensemble method for classification. The ensemble is a combination of decision trees built from a bootstrap sample from training set. Additionally, in building the decision tree, the split which is chosen when splitting a node is the best split only among a random set of features. This will increase the bias of a single model, but the averaging reduces the variance and can compensate for increase in bias too. Consequently, a better model is built. In this work, the implementation of random forests in scikitlearn python library is used, which averages the probabilistic predictions. Two number of estimators are simulated for this method. With 10 estimators, the overall error is 2.16%, SC is 87.7 %, and BH is 0.73%. Using 100 estimators will result in overall error of 1.41 %, SC of 92.2 %, and BH of 0.51 %.

We observe that comparing to the naive Bayes algorithm, although the complexity of the model is increased, yet the performance does not show any improvement.

4.5 Support Vector Machines :

Linear kernel gains better performance compared to other mappings. Using the polynomial kernel and increasing the degree of the polynomial from two to three shows improvement in error rates, however the error rate does not improve when the degree is increased further. Finally, applying the sigmoid kernel results in all messages being classified as hams.While the overall training set error of the model is far less than error rate for naive Bayes, the test set error is well above that rate. This characteristic shows the model might be suffering from high variance or overfitting on the data. One option we can explore in this case is reducing the number of features. However, the simulation results show degradation in performance after this reduction. For instance, choosing 800 best features based on MI with the labels and training SVM with linear kernel on the result yields to 1.53% overall error, 91.5% SC, and 0.53% BH. While applying SVM with different kernels increases the complexity of the model and subsequently the running time of training the model on data, the results show no benefit compared to the multinomial naive Bayes algorithm in terms of accuracy..

4.6 Adaboost:

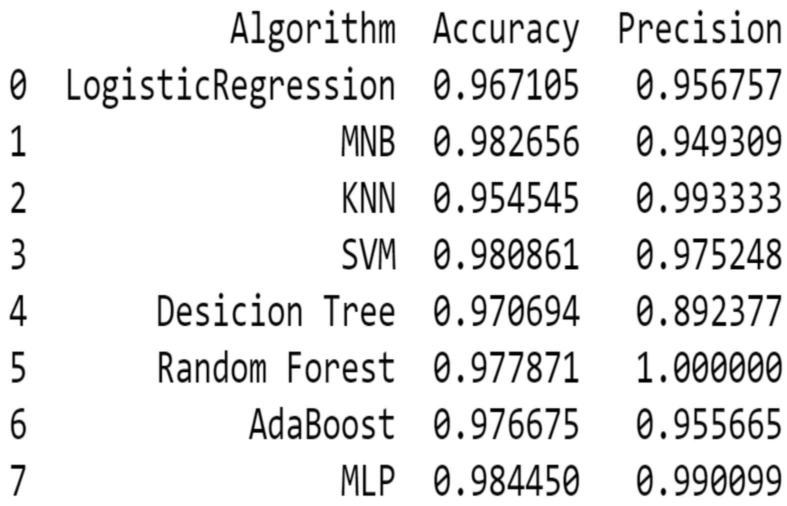
Adaboost is a boosting ensemble method which sequentially builds classifiers that are modified in favor of misclassified instances by previous classifiers . The classifiers it uses can be as weak as only slightly better than random guessing, and they will still improve the final model. This method can be used in conjunction with other methods to improve the final ensemble model. In each iteration of Adaboost, certain weights are applied to training samples. These weights are distributed uniformly before first iteration. Then after each iteration, weights for misclassified labels by current model are increased, and weights for correctly classified samples are decreased. This means the new predictor focuses on weaknesses of previous classifier. We tried the implementation of Adaboost with decision trees using scikit-learn library. Like Random Forests, although the complexity is much higher, naive Bayes algorithm still beats Aadaboost with decision trees in terms of performance.

4.6 Multi Layer Perceptron:

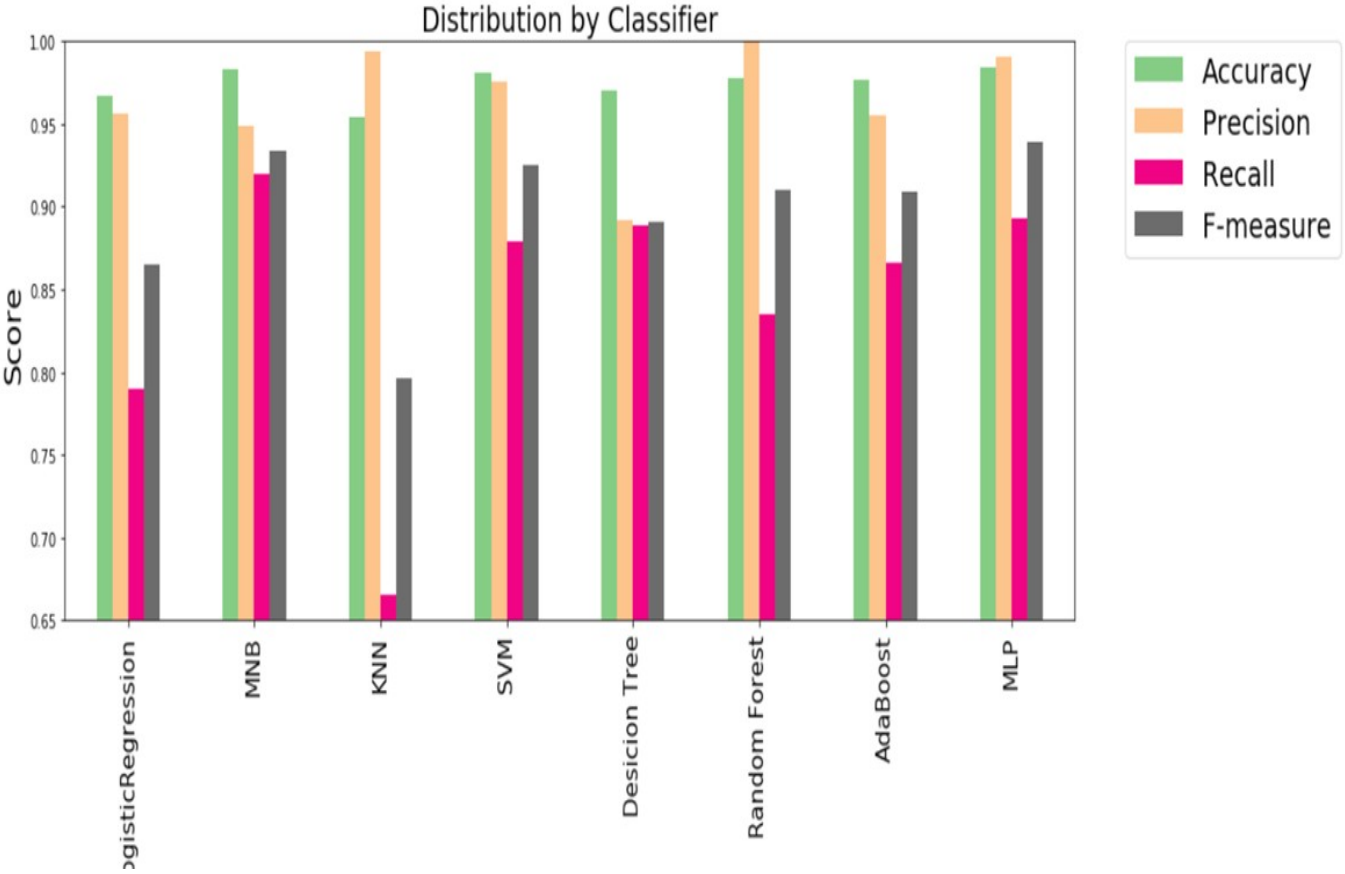
A multilayer perceptron (MLP) is a class of feedforward artificial neural network (ANN). The term MLP is used ambiguously, sometimes loosely to any feedforward ANN, sometimes strictly to refer to networks composed of multiple layers of perceptrons (with threshold activation).

An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training.[2][3] Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can distinguish data that is not linearly separable.

5. Result:



```
            Algorithm  Accuracy  Precision
0  LogisticRegression  0.967105   0.956757
1                 MNB  0.982656   0.949309
2                 KNN  0.954545   0.993333
3                 SVM  0.980861   0.975248
4       Desicion Tree  0.970694   0.892377
5       Random Forest  0.977871   1.000000
6            AdaBoost  0.976675   0.955665
7                 MLP  0.984450   0.990099
```



6.Conclusion:

The recent surge of mobile phone use makes the emerging communication media such as SMS articularly attractive for spammers. The challenge of detection spams in SMS is due the small number of characters in short text messages and the common use of idioms and abbreviation. The research that does exist on SMS spam detection has only focused on word distribution but has yet to examine explicit semantic categories of text expressions.

Content based filtering suffers from challenges like short content, abbreviated words and user content safety. All of the studies tried to solve some challenges of SMS spam detection. Bayesian algorithm also suffers from traditional threshold selection problem, dataset dependency, assuming prior probability. Despite having those shortcomings, Bayesian is declared as the most suitable algorithm for spam filtering.

In the current study, we proposed to employ categories of lexical semantic features in the detection of SMS spam. Our experiment results show that incorporating semantic categories improve the performance of SMS spam detection.The features identified in this study may be applied to improve spam detection in other types of communication media such as emails, social network systems, and online reviews. These features will also help to improve mobile users' aware of spam and their knowledge on how to detect spams in SMS.

There are some interesting future research issues such as incorporating dynamic features by tracking the usages of different words over time and testing the generality of the proposed features in other communication media such as online review and social networks. In addition, there is space for further improving the performance of spam detection.